Faces In Places: Compound query retrieval

Yujie Zhong, Relja Arandjelovic, Andrew Zisserman

Faces In Places: compound query retrieval

Yujie Zhong[1]
yujie@robots.ox.ac.uk
Relja Arandjelović[2]
relja.arandjelovic@inria.fr
Andrew Zisserman[1]
az@robots.ox.ac.uk

[1] Visual Geometry Group
Department of Engineering Science
University of Oxford, UK

[2] WILLOW project
Departement d'Informatique de l'École Normale Supérieure
ENS/INRIA/CNRS UMR 8548

Abstract

The goal of this work is to retrieve images containing both a target person and a target scene type from a large dataset of images. At run time this *compound query* is handled using a face classifier trained for the person, and an image classifier trained for the scene type. We make three contributions: first, we propose a hybrid convolutional neural network architecture that produces place-descriptors that are aware of faces and their corresponding descriptors. The network is trained to correctly classify a combination of face and scene classifier scores. Second, we propose an image synthesis system to render high quality fully-labelled face-and-place images, and train the network only from these synthetic images. Last, but not least, we collect and annotate a dataset of real images containing celebrities in different places, and use this dataset to evaluate the retrieval system. We demonstrate significantly improved retrieval performance for compound queries using the new face-aware place-descriptors.

1 Introduction

Suppose you want to find that photo of your friend in the cathedral in your personal collection of 100k images, or you are a news producer and want to find a photo of "Barack Obama on the beach" to illustrate an article. Searching large scale image collections visually for *compound queries* of this type is still a challenging and little researched problem. In contrast, due in part to the success of classifiers based on deep convolutional networks [11, 20], there has been tremendous progress in search and annotation for *single queries*, such as finding particular people by their face [5, 13, 18, 23], or specific objects [2, 10], object categories [4, 16], or scene categories [25].

The objective of this paper is high performance (precision and recall) retrieval for compound queries consisting of a 'face' and a 'place' in large scale image datasets. A naive approach to this problem would be to train a classifier for the face of interest, say 'Anne Hathaway', and the place, say a supermarket, obtain a ranked list of images for each based on the classifier score, and then combine the lists in some way using the rankings or scores [2, 19]. However, such combinations often have poor performance as an image with a high classifier score for a place, may have a very low score (and so low rank) for a face, and vice versa (for example, images that contain very recognizable faces usually have a large face proportion and much less information about the background scene). Instead of trying to find the best

combination method for such independent and incomparable scores, we *choose* a combination rule, and then design and train a deep convolutional neural network (CNN) to optimize the classification score for this rule. This formulation of the problem as training a CNN to achieve a common classification is one of the key contributions of this paper.

As will be seen in the sequel, training the CNN to generate place-descriptors (feature vectors) for a common classification with face-descriptors encourages the place-descriptors to be 'face aware', in the sense that they are able to ignore the face regions in an image and concentrate on the class of the place information. They are also 'face-descriptor aware' as they are calibrated to cooperate with the face-descriptors.

To test our method, and to facilitate research in compound query retrieval, we build a 'Celebrity in Places' (CIP) dataset, consisting of images downloaded from the Internet and manually annotated using Mechanical Turk (section 4). However, it is not possible to gather training data in the same manner because the yield is too small. Furthermore, achieving a good coverage of faces in places is impossible as images of many face-place combinations simply do not exist – *e.g.* searching the Internet for "Angela Merkel in an ice rink" gives no relevant images. To alleviate this problem, we develop an image generation pipeline to automatically synthesize high quality images for any face-place combination (section 5). The CNN is trained using only these synthetic images. We show in section 6, that the face aware place-descriptors lead to a substantial improvement in compound query retrieval performance, compared to baselines with state-of-the-art descriptors used independently.

2 Retrieval for Compound Queries

Given *any* face-in-a-place compound query, our goal is to rank images containing that face-in-a-place combination highly. For the retrieval system to be useful, we would like to be able to answer queries which have not been seen at the training stage, containing potentially novel faces and novel places. Therefore, standard approaches such as pre-tagging the database with a fixed 'vocabulary' of concepts (here faces/places), as commonly done for the TRECVID semantic indexing challenge [14], is inappropriate.

To this end, we represent each database image using a set of feature vectors, one for the place and one for each face in the image. Given a new compound query, a classifier is learnt online which is then used to rank the images. Since it is very unlikely that we can obtain a sufficient amount of training images depicting the compound query ('Barrack Obama on the beach'), but it is easy to obtain images of the face ('Barrack Obama') and the place ('beach'), the face and place classifiers are trained independently. They are then used to obtain two scores for each database image, one for a face and one for a place, and these scores are combined and sorted to obtain the final ranked list. Since the task is to find the face *and* the place, a natural way to combine the scores is to take their minimum, as this penalizes non-existence of any of the two objects of interest. Clearly, the two scores have to be properly calibrated in order for this operation to make sense – the calibration is explicitly addressed in our network architecture (section 3).

In terms of efficiency, since linear (SVM) classifiers are used and the image descriptors are pre-computed, images can be classified and ranked for compound queries using standard methods [4, 10] which enable image datasets of millions of images to be searched in a fraction of a second.

3 Network Architecture

Our objective is to train a network which produces a better feature representation for places by making them interact with the face-descriptors. The new descriptors should serve two

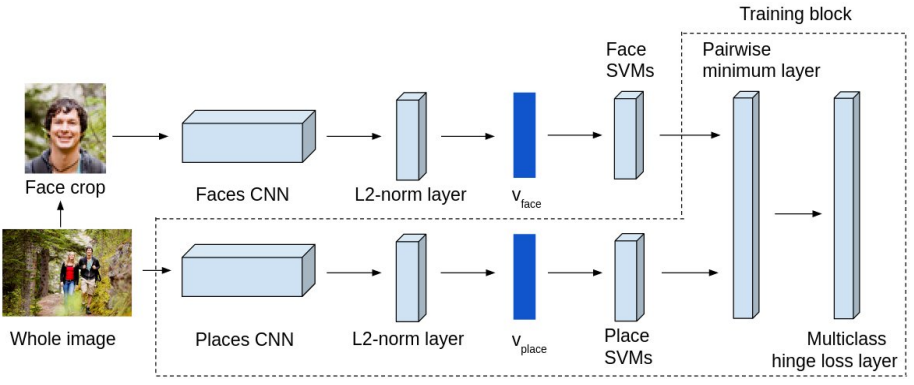

Figure 1: **Hybrid network architecture.** The network is used to generate face-descriptors and face-aware (FA) place-descriptors (the vectors v_{face} and v_{place}) for the face crop and the whole image respectively. These descriptors are used to represent the images in the target dataset. The operation of the network is to first compute FC7 feature vectors in each stream for the face crop and whole image respectively; then the vectors are L2-normalized to form v_{face} and v_{place}. The network is trained with additional layers (right of the feature vectors) that mimic the query-time operation of ranking the dataset images using linear classifiers (SVMs) and combining the face and place classifier scores by taking their minimum. During training, parameters of all the layers within the dotted block are updated.

major purposes – firstly, place descriptors should be "aware" of the existence of faces, and the place descriptor should "suppress" the face information and focus mainly on the other areas of the image. Secondly, the place descriptors should be "aware" of the face *descriptors*. That means the descriptors should be amenable to the combination rule we chose – the relevance of a database image to the compound query is computed as the minimum of the query face and query place classification scores (section 2). By explicitly incorporating this rule during training, the place-descriptors can be learnt such that the classification scores are calibrated.

The coupling of the face and place descriptors, addressing both aspects of the new features, is achieved using our hybrid network, described next.

3.1 Hybrid network

Here we introduce the hybrid network architecture, shown in figure 1, and explain how it addresses the two challenges that the place-descriptor must satisfy described above. The training of the network is described in Section 3.2.

Two streams. The network has two streams, one for the place and one for face descriptor extraction. The place stream is an AlexNet pre-trained on the Places205 dataset [25], and the face stream is a VGG-16 very-deep network trained on the VGG Face Dataset [13]. The two networks are cropped at layer FC7 and the outputs are L2-normalized to form the descriptors, v_{face} and v_{place}. These descriptors are used to train the face and place SVM classifiers.

SVM classification layer. The normalized descriptors are passed through the classification layers independently. The two classification FCs are constructed by stacking binary SVMs. Therefore, the size of the place SVM layer is $N_p * L_p$, where N_p is the number of place classes and L_p is the length of v_{place}, and similarly for the face SVM layer. During training, the place SVM layer is updated by the loss back-propagated from the layer described next.

Pairwise-minimum layer. On the top of the network, the scores for the face and place classes are combined into the scores for the face-place "product" classes, i.e. scores for all

face-place combinations. For example, if there are 100 face classes and 10 places classes, then there are 1000 product classes. In order to mimic the query-time operation, the score of a face-place combination is the minimum of the scores for the two classes. Finally, the face-place scores are passed through a loss layer, where a multiclass hinge loss proposed by Crammer and Singer [6] is applied. Therefore, the two streams are actually coupled by the pairwise-minimum layer. The reason for using the multiclass hinge loss instead of logistic loss is because we aim to learn the place CNN such that the place classification layer acts as a stack of binary SVMs, which exactly matches the case during query time.

Design choices. During training, the place stream is updated, whilst the face stream is fixed. The reason for learning the place stream only is because the face descriptors are only computed from a detection window around the face, and so are less influenced by the scene. In contrast, the place descriptors are extracted from the entire image and so can take into account that there should be a face in the image, and that a person will occlude a part of the scene. Moreover, the calibration can still be learnt between face and place descriptors even though only the place descriptors are updated during training.

3.2 Implementation Details

Faces are detected using the method of [12]. The Place-CNN and Face-CNN both produce FC7 feature vectors vectors that are 4096 dimensional. The output of the pairwise minimum layer is a vector of length $N_f * N_p$, where N_f and N_p are the number of face and place classes respectively in the training set.

Training SVM classifiers. The linear SVM classifiers (for each face and place class) are trained in a one-vs-rest manner, where descriptors from all the other classes are the negatives. A weighted loss is used to balance the positive and negative sets. The SVM weight vectors are L2-normalized. These classifiers are used to rank the test images at query time, and are also used to initialize the SVM layers for training the network. This is important, as it leads to faster convergence.

Network training. Augmentation is used in both the face and place stream. For the place stream, the original full size image is resized so that its shorter side is 256 pixels, and a 227 pixel crop randomly selected (such that the crop contains the target face). The face stream is similarly resized, but the crop size is 224 pixels. Both the original place image and its face crop are flipped horizontally with 50% probability. The place SVMs in the classification layer are L2-normalized at each gradient descent step to mimic the query time situation where the SVMs are L2 normalized. The network is trained for 5 epochs using dropout at a rate of 0.5 for the FC layers. Each epoch takes about 3 hours using a single GPU, with an average training speed of 18 images/sec. MatConvNet [21] is used for the implementation.

4 The "Celebrity in Places" (CIP) Dataset

In order to evaluate the compound query approach in a real world, images containing celebrities in places are downloaded from the Internet. These images form our retrieval test set. Figure 2 shows examples from this "Celebrity in Places" dataset, which contains 4611 celebrities. In this section we describe how this dataset was built by a combination of web search and Mechanical Turk annotation.

The dataset is obtained in three stages: (i) initial downloading using Google Image Search based on a text query; (ii) duplicate removal; and (iii) manual checking using Mechanical Turk.

Figure 2: **Example images from the CIP dataset.** The celebrities include David Cameron, Emma Watson, Pau Gasol etc., and the places cover airport terminal, office, etc.

The query texts used for the Google Image Search are the pairwise combinations of a celebrity name list and a places list. The celebrity name list is compiled from two sources. The first list has 2622 celebrities from the VGG Face Dataset [13]. The second contains 1989 celebrities that are not included in the VGG Face Dataset, but are nonetheless very famous (these celebrities were excluded from the VGG Face Dataset to avoid overlap with prior face datasets such as LFW [8] and YTF [22]). The scene list contains 16 place classes selected from the Places205 Dataset [25] classes. These classes are the ones for which celebrities are more likely to have photos taken (celebrities are very unlikely to be photographed in some places, such as computer room or creek). They are airport terminal, banquet hall, boat deck, coffee shop, conference room, desert, golf course, hospital room, ice skating rink, kitchen, ocean, office, staircase, supermarket, stadium and stage (and synonyms of these).

However, the returned images for a text search composed of a person name and a place are very noisy – far more so than searches on the individual name or place. This is probably a reflection of the lack of joint face-place annotation in the caption of many images. Also, the downloaded dataset contains many duplicates. To alleviate this duplicate problem, exact and near duplicate images are removed by clustering VLAD descriptors [3, 9] of all images, and only retaining one image per cluster.

Only a small fraction of the downloaded (unique) images actually contain the named person in the specified place. Mechanical Turk (MT) is used to obtain the images that do. To avoid checking all images, we follow the annotation procedure suggested in [13]: first, CNN image classifiers are trained for each of the 16 place classes (using images from the Places205 dataset), and used to rank the downloaded images. Then only images of that place-class above a classifier threshold are sent to the annotators. This procedure substantially reduces the number of images that need to be annotated, and hardly misses any positives.

Over 2.5 million images are downloaded, 170k remain after refining by pre-trained CNN and deduplication, and they are sent to the MT annotators. Only 38k images survive. This is a very low yield, and exemplifies how noisy the original downloaded images are. The 38k images includes 4611 different celebrities and 16 places. The distribution histograms for both people and places are included at [1] .

5 Automatic Generation of Synthetic Training Images

We generate a synthetic training dataset for two main reasons: first, it is difficult to obtain a sufficient number of high quality training images using Image Search Engines; second, those images that are obtained are highly unbalanced across classes. These two problems, which were described further in Section 4, can severely negatively impact the CNN performance. and hence the performance of the network. On the other hand, we can, to a large extent,

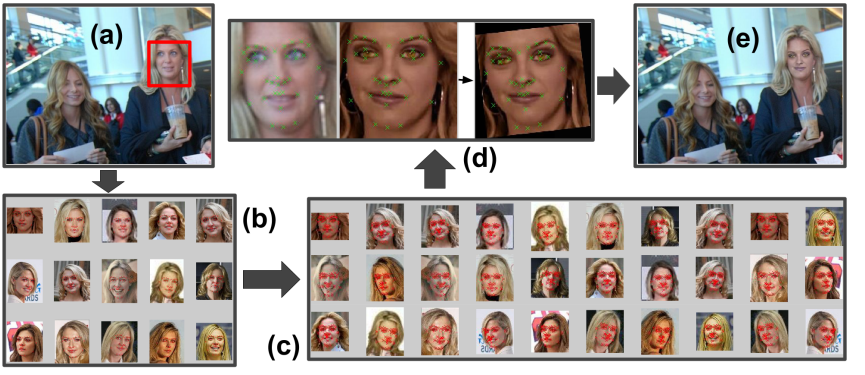

Figure 3: **Automatic synthetic rendering pipeline.** Replacing the face of the unknown person in the airport with the celebrity face (Gage Golightly). (a) Face detection and 36 landmark points annotation on the target image. (b) Top K most similar images (based on FC7 feature vectors) in the face source dataset. (c) Pose descriptors are computed based on face landmarks for these top K images and their flipped version. Top N images in the re-ranked list based on pose similarity are qualified as source faces for replacement. (d) During face replacement, a similarity transformation is computed between corresponding landmark points to map the source face onto the target one. (e) Poisson Editing is applied to produce natural-looking synthetic images. At this stage, the original face is replaced by the labelled source face in the image.

control these two factors if we synthesize our training dataset. Since it is straightforward to obtain images labelled with places (from the Places205 [25] dataset), our goal is to replace the unknown faces in place-labelled images with labelled celebrity faces; since then both place and face identity are labelled for that image. The challenge is to replace the face without introducing artefacts. In this section, we describe the face replacement method (shown in Figure 3), and give statistics of the generated dataset.

5.1 Source and target datasets

A "target" image is the image whose place label is known but which contains unknown faces. The unknown faces will be replaced with known faces coming from a "source" image, thus creating an image with complete face and place labels.

Target places dataset. The Places205 [25] dataset fits our requirements for the target dataset perfectly as it contains a large number of scene-centric images, and is fully labelled. In order to be able to replace faces, images which do not contain a face are filtered out. We keep the same 16 place classes as for the CIP dataset (c.f. section 4).

Source faces dataset. The VGG Face Dataset [13] is the largest publicly available face dataset, containing 2622 face classes with a plentiful number of samples for each class. We use the 500k manually annotated subset of this dataset as source face images which will be pasted into the target images.

5.2 Automatic Face Replacement

We wish to replace a face in a target image with a face sourced from another image. There are two steps: (i) selecting candidate faces from the 500k potential images, and (ii) replacing the target face with the source face.

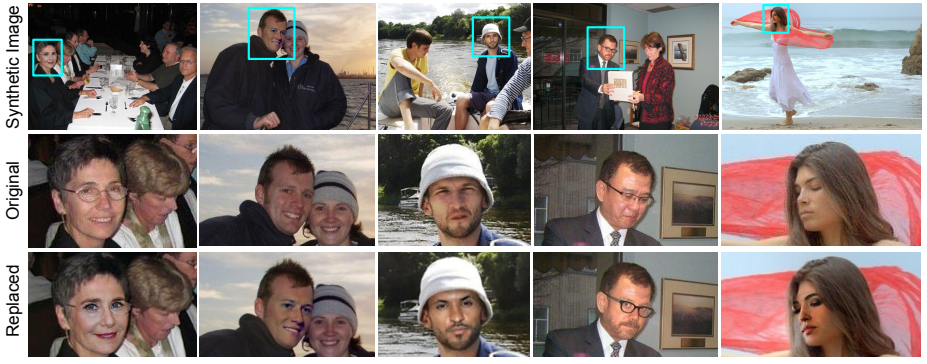

Figure 4: **Example images from the synthetic dataset.** The first row shows the synthetic images and the replaced faces inside the blue boxes. The second and the third rows show close-ups of the original and the replaced faces, respectively. The automatic face search and replacement system successfully deals with different lighting condition and head poses, as well as accessories like hats and glasses. Further examples are given at [1].

For (i) the aim is to select candidates that have the most similar appearance and pose to the target face, as smaller changes reduce the risk of introducing artefacts. This selection proceeds in three stages as illustrated in figure 3: **(1)** find the top K appearance-similar source faces, where appearance-similarity is measured using the 4096 dimensional CNN face-descriptor extracted from FC7 layer, i.e. we search for faces that could be confused with each other for recognition. **(2)** Rank these K candidates by their similarity in pose, where pose is measured by the Euclidean distance between every pair of facial keypoints of each face. These 36 keypoints are computed using the algorithm of [24]. In practice this delivers source faces that have similar shape and skin colour to the target. **(3)** Bad quality face crops are removed from the candidate list to ensure the quality of the source faces. Basically, source face candidates must be correctly predicted by Deep Face CNN with its probability over a certain threshold.

Given a candidate source face, the replacement (ii) also proceeds in three stages: **(1)** using the 36 keypoints, find the transformation between the source and target face. We compute a similarity transformation as this is less distorting than an affine one (and the face poses are close). **(2)** blend the source face into the target image. This is achieved using Poisson Editing described by [15]. **(3)** Check that the face can be recognized using a face classifier trained for that identity. This is a check that no artefacts have been introduced that can hurt the recognition of that face. The synthetic image is preserved only if the replaced face has a classification score higher than a threshold. The result of this pipeline is that the source face is blended in quite naturally to the target image. Some examples are shown in Figure 4.

5.3 Synthetic dataset statistics

The dataset produced has 178k training and 8.7k validation images (out of 200k images synthesized, with only images above a reasonable CNN score threshold retained). The images are class balanced for face classes as this synthetic dataset is generated under certain rules to ensure the maximum number of images per face class and the diversity of face classes for each target image. The training and validation set share the same 500 face classes and

Test set name	Query face classes		Target images		Distractors	Total
	unseen	seen	unseen	seen		
Synthetic	100	500	7.6k	8.3k	58k	73.8k
Celebrity in Places	792	223	1.7k	0.6k	58k	73.1k

Table 1: **Test sets statistics.** "Seen" and "unseen" correspond to query face classes which have or have not been seen by the network at training time, respectively. Target images are the set of all images which are positive for some query, while distractors are images which are negatives for all queries. For each query, there are at least 1 and at most 21 target images.

16 place classes with a similar distribution, while target images in our synthetic test set (introduced in Section 6) have 100 faces classes that the CNN has never seen during training. However, there are no instances in common at all between the training and validation sets (i.e. no target place-images or source faces). This synthetic dataset is significantly larger than the downloaded Celebrity In Places Dataset described in the previous section.

6 Experimental Evaluation

In this section we describe the test datasets, the evaluation protocol, and the baselines, followed by quantitative and qualitative evaluation.

Test Datasets. We test the compound query retrieval on two test sets – our new "Celebrities in Places" (CIP) dataset (section 4), and the test subset of the synthetic dataset (section 5). To provide more detailed results, we divide the queries for each test set based on whether the query face class has been seen at training time or not. Namely, the performance on "seen" classes demonstrates the ability of the system to search for faces which are already known to the system (though the specific face instances have not been used at training), while good performance on the "unseen" queries provides a stronger proof of generalization of the system's ability to answer any unconstrained query. To make the tests more challenging, we also augment each dataset with real distractor images from the Places205 dataset; the distractors do not contain any celebrities and therefore are negatives for all test queries. The same 16 place classes are used throughout. The statistics of the two test sets are shown in table 1.

Evaluation Protocol. To evaluate the quality of compound query retrieval, for a given query we consider as positives only images that contain both the query face and place, and all other images are considered as negatives, even if they contain the query face or query place. For a given query, we compute Average Precision (AP) as well as recall at rank 5 (i.e. recall is deemed to be one if a positive is retrieved within the top 5 ranked images, otherwise 0), and these are averaged across queries to obtain the mean Average Precision (mAP) and the mean recall@5. To disentangle the quality of face and place retrieval, we also report the individual mAPs for faces and places separately.

Baselines. We compare to three late-fusion approaches that use independent face and place descriptors, instead of our face-aware place descriptors. The baselines differ in the Place-CNN used, and in the method of calibration. The first, *Places-L2norm* uses the Place-CNN trained on the entire Places205 dataset, with calibration by L2-normalising the weight vector for both the place and face SVMs (before combining the scores using the minimum score rule). The other two baselines are stronger – the Place-CNN is retrained to recognize only the 16 place classes using only the images in the Places205 dataset that contain faces (this improves the performance of scene classifications for images containing people). These two baselines differ only in the method of calibration: one, *Places-F-Platt*, is calibrated using

Platt [17]; the other, *Places-F-L2norm*, is calibrated using L2-normalization of the weight vector.

6.1 Retrieval results

The retrieval performance is given in Table 2 and retrieval examples are shown in Figure 5 (there are further examples at [1]). As is evident from the results, our new place descriptors perform much better than the three baselines. The best baseline is *Places-F-L2norm* for both 'faces in places' and 'places only'. As expected, finetuned Place-CNN (*Places-F-L2norm* and *Places-F-Platt*) have a higher 'places only' mAP than the original Place-CNN (*Places-L2norm*). Furthermore, the fact that *Places-F-Platt* performs worse than *Places-F-L2norm* for 'faces in places' task indicates that L2-normalization of the classification weight vectors is a better calibration method than Platt-scaling. Notice, for the 'places only' task, our models achieve a higher mAP (0.514) than the Places205 CNN specifically finetuned for the 16 place classes images with faces (0.441). This supports our argument that it is not just calibration that is being learnt, it is also a better face-aware place descriptor.

For the 'faces in places' task, the performance gap between the baselines and our method becomes more significant when testing on the real target images in the CIP test set. Most importantly, our descriptors generalize well to work with the 'unseen' face classes, and achieve a very high Rec@5 (0.807 and 0.760) for both test sets. Finally, we can improve the performance further with augmentation (mAP of 0.675 and 0.593) by taking the mean descriptor from 10 crops for faces.

test set	descriptors	**faces in places**		faces only		places only
		unseen	seen	unseen	seen	
Syn	Places-L2norm	0.480 / 0.874	0.363 / 0.569	0.899	0.692	0.521
	Places-F-L2norm	0.533 / 0.859	0.426 / 0.649			0.534
	Places-F-Platt	0.491 / 0.830	0.330 / 0.557			
	Hybrid CNN	0.655 /0.941	0.617 / 0.817			**0.604**
	Hybrid CNN-Aug	**0.676 / 0.946**	**0.655 / 0.846**	**0.914**	**0.743**	
CIP	Places-L2norm	0.381 / 0.550	0.325 / 0.502	0.693	0.699	0.381
	Places-F-L2norm	0.435 / 0.605	0.373 / 0.574			0.441
	Places-F-Platt	0.420 / 0.609	0.239 / 0.388			
	Hybrid CNN	0.640 / 0.807	0.577 / 0.760			**0.514**
	Hybrid CNN-Aug	**0.675 / 0.867**	**0.593 / 0.777**	**0.741**	**0.737**	

Table 2: **Retrieval mAP and Recall@5 on the test sets.** 'Aug' means that an augmentation is applied on the face feature vectors by taking the mean of 10 crops (single values are mAP only).

In general, images of supermarkets, beaches and stages are well retrieved by compound-queries, whereas others are sometimes confused (*e.g.* football stadium and golf course are confused due to the large green areas). Furthermore, the face-aware place descriptors have a much better performance than the baselines when faces do not take up too much of the image (when they are too large, the background provides too little information on the scene).

To illustrate the difference between the face-aware and original (i.e. trained on Places205 Dataset) place descriptors, we search for nearest neighbours in the entire Places205 Dataset using only the place descriptors. As shown in Figure 6, when querying with an image that contains faces, the most similar images returned by the face-aware descriptors do not necessarily contain large faces but do match the place class. In contrast, the original place descriptors find scenes containing faces, but often with the wrong place class. This reveals that the new descriptors tend to ignore the face and focus more on the background information, and consequently are more accurate for the compound query.

Figure 5: **Examples of the top two retrieved images for various compound queries on the CIP test set.**

Figure 6: **The top ranking images by nearest neighbour search on place descriptors.** For each query image, the upper row uses the original Place-CNN descriptors, whilst the lower row uses face-aware descriptors. Left: airport terminal, right: boat. The green box indicates the same scene as the query image, whereas red indicates a different scene.

7 Summary

We have proposed a compound query retrieval method which searches for particular people in particular scenes in a large image dataset. We also present a hybrid network in which faces and places pre-trained networks generate place descriptors that are aware of faces and face descriptors. In order to train the proposed network, we have designed an automatic pipeline to synthesize high quality training images. Finally, we collected a Celebrity in Places (CIP) Dataset to evaluate our methods, and demonstrate that we outperform the baselines significantly. The CIP Dataset is available at [1].

Of course, the methods developed are agnostic about the particular people and places, and can be applied to personal image and video collections, not just to celebrities in places. Also, the hybrid CNN architecture is independent of the underlying face and place CNNs used, and further performance boosts can be expected by replacing those CNNs with more powerful CNNs (such as ResNet [7]) in the future.

Acknowledgements. Financial support was provided by the EPSRC Programme Grant Seebibyte EP/M013774/1. Relja Arandjelović is supported by the ERC grant LEAP (No. 336845).

References

[1] Research page. http://www.robots.ox.ac.uk/~vgg/research/faces-in-places/.

[2] R. Arandjelović and A. Zisserman. Multiple queries for large scale specific object retrieval. In *Proc. BMVC.*, 2012.

[3] R. Arandjelović and A. Zisserman. All about VLAD. In *Proc. CVPR*, 2013.

[4] K. Chatfield, R. Arandjelović, O. M. Parkhi, and A. Zisserman. On-the-fly learning for visual search of large-scale image and video datasets. *International Journal of Multimedia Information Retrieval*, 2015.

[5] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High dimensional feature and its efficient compression for face verification. In *Proc. CVPR*, 2013.

[6] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *JMLR*, 2:265–292, 2001.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[8] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[9] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Proc. CVPR*, 2010.

[10] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local images descriptors into compact codes. *IEEE PAMI*, 2012.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[12] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistle. In *ECCV*, 2014.

[13] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *Proc. BMVC.*, 2015.

[14] O. Paul, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, and G. Quéenot. TRECVID 2011 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID*, 2011.

[15] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM Transactions on Graphics (TOG)*, volume 22(3), pages 313–318. ACM, 2003.

[16] F. Perronnin, Z. Akata, Z. Harchaoui, and C. Schmid. Towards good practice in large-scale learning for image classification. In *Proc. CVPR*, pages 3482–3489, 2012.

[17] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods - Support Vector Learning*, pages 185–208, 1999.

[18] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proc. CVPR*, 2015.

[19] J. A. Shaw and E. A. Fox. Combination of multiple searches. In *The Second Text REtrieval Conference (TREC-2)*, pages 243–252, 1994.

[20] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[21] A Vedaldi and K Lenc. Matconvnet – convolutional neural networks for matlab. In *Proc. ACMM*, 2015.

[22] L. Wolf, Tal. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched backgroundsimilarity. In *Proc. CVPR*, 2011.

[23] Z. Wu, Q. Ke, J. Sun, and H. Shum. Scalable face image retrieval with identity-based quantization and multireference reranking. *IEEE PAMI*, 33(10):1991–2001, 2011.

[24] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *Proc. CVPR*, 2013.

[25] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, pages 487–495, 2014.